Airport Departure Planning and Control

A Departure Planner Decision-Aiding System
Runway Sequencing & Scheduling
Modeling Approaches

Ioannis Anagnostakis
Prof. John-Paul Clarke

International Center for Air Transportation
Massachusetts Institute of Technology

JUP meeting, NASA Ames
Thursday, April 04, 2002

Background & Motivation

- Controllers usually handle high volumes of airport traffic very well, however...
  - System state (aircraft and vehicle position and intentions) is highly unpredictable and only partially observable to controllers
  - Delays emanating from the complex nature of airport operations are still ailing the system
- Uncertainty manifests in the form of congestion queues
  - Significant operational delays observed in the departure process
  - Excessive fuel burn
  - Considerable environmental impacts associated (aircraft emissions,noise)
- Need to design and implement a decision-aiding system to:
  - Control airport congestion (network of queues)
  - Exercise tighter sequencing and scheduling control on each portion of the departure process
- With the objective to:
  - Mitigate existing inefficiencies (improve system capacity utilization) and reduce observed delays
  - Assist air traffic controllers in enhancing the performance of departure operations
- Thorough understanding of the links and interactions in ATM operations required

Approach

- Identify the dynamic behavior of the airport system:
  - Understand system dynamics through field **observations** and data collection (BOS, EWR, IAD)
  - Identify system inefficiencies, flow constraints, interactions between them and their causal factors
  - Study the propagation and effects of constraints Identify control points, flow control options and controller strategies
- **Formulate** the problem
  - Define and prioritize system-wide objectives and design objective function
  - Incorporate system constraints in the problem formulation
- Propose a system **architecture**
  - Conceptual design for the DP decision aid to operate in a multi-objective environment, with (often) competing system-wide objectives
  - Virtual Queue Manager as the system coordinator
- Design **optimization** logic and strategies
- Investigate implementation and human factors issues

Field Observations: Airline Station Visits

- BOS
  - Proximity
  - Close relationship with FAA and airlines (UAL, USAir visits)
  - Severe ramp and taxiway space constraints
- EWR
  - Case study based on collected data
  - CO visit
- IAD
  - Baseline case data on how the system operates
  - UAL station visit
- Understand decision processes: Airline, Tower & TRACON, FAASCC
- Investigate the differences in objectives between the airlines and the FAA
- Constraint propagation from the SCC to the airport surface

Airline and Airport Operational Issues

- Ramp Coordination
  - Who should be in control and why ?
  - What should be under the control of each party involved (airlines, airport authority, FAA Tower) ?
  - Should there be shared responsibility between the airlines and the tower ?

- Information
  - Known vs. Necessary but not known
  - Facilitate information exchange

- What constraints have what impact ?
  - Center, Tower or SCC generated constraints and how do they force the shutdown or restriction of certain departure flows

- Constraint/Restriction triggering
  - For each position in the SCC, what is their basis for triggering constraints ?
  - How do they decide on the level of the restriction ?
  - Buffering: What can make the decisions be less conservative (ADS-B, GPS)
  - How are holding decisions made ?? Where and for how long ??

- Airlines
  - Are there benefits in prioritization ?
  - N-control and VQ integration issues
  - Individual’s performance metrics (who is held responsible in extreme cases ?)

Field Observations (BOS): Control Points and Control Functions

- A control point is the last chance to apply a particular control function to the departure queues
  - Physical point on airport surface or point in time when aircraft transitions from one state to another
- Main departure flow control points identified, are:
  - Gates / Ramp
  - Point of entry into taxiway system (from the gate or ramp)
  - Point of commitment to specific queue (temporal or spatial)
  - Point of entry to active runway (from a takeoff queue)
- Control Functions are:
  - Queue Size Control: Pushback clearance (jets) or taxi clearance (props) (Gates)
  - Engine Run Time Control: Engine Start Control (Gates / Ramp)
  - Runway allocation and taxi-path control: Routing aircraft to a specific runway
  - Taxi Time Control: Clearance to enter the taxiway system from ramp area or gate
  - Takeoff Sequencing: Merging of aircraft into the same takeoff queue or mixing between aircraft from multiple queues
  - Takeoff Release: Involves mixing of operations on runways used by departures, landings and runway crossings

Field Observations (BOS) (cont.)

- Airport is an interactive queuing system => control strategies focus on queue mgt
- The flow constraints identified were associated with the main airport system elements:
  - Runway system (Key Flow Constraint)
  - Gates, Ramps and Taxiways (Secondary Flow Constraints)
- Main Causal Factors for the Flow Constraints were Identified
- Controller Workload Introduces Additional Flow Constraints

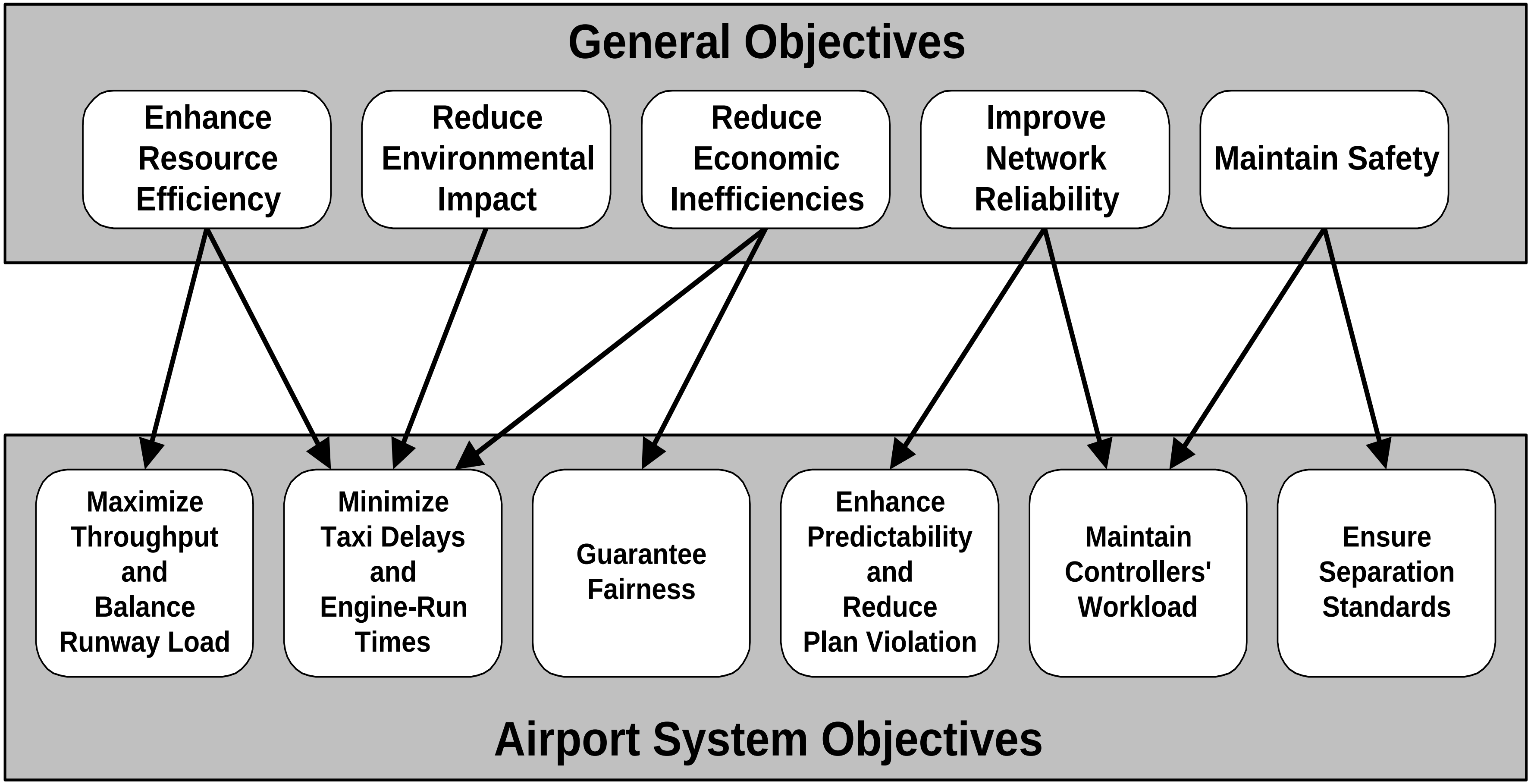
  (Parallel Flight Strip Queuing Process in the Tower)
- Boarding and pushback procedures are not very observable and are highly volatile
- Ample opportunity to affect the final runway operations sequence
- Major importance given to downstream constraints

Note: Results reported in GNC 98 and ATM 98 conference papers

Field Observations: Identified Inefficiencies

- Wake vortex separation restrictions (weight class mixing)
  - Can be waived by some pilots though, differ from airport to airport
- Downstream constraints
  - Time window (EDCT)
  - Spacing
    - ✓ Departure fix congestion (jet / prop mixing, arr. & dep. mixing, splitting departures)
    - ✓ MIT, DSP
  - Delay (GS, GDP)
- Runway operations mixing
  - Regular and midfield departures
  - Arrivals
  - Crossings
- Takeoff buffer size
  - Unnecessarily large: increased taxi delays
  - Very low: runway “starvation”
- Intersecting runways (LAHSO-Land And Hold Short Operations)
- Restricted runway access: inability to take off & recovery effort
- Controller Workload Constraints

Problem Formulation: Objectives



General Objectives

Enhance Resource Efficiency

Reduce Environmental Impact

Reduce Economic Inefficiencies

Improve Network Reliability

Maintain Safety

Maximize Throughput and Balance Runway Load

Minimize Taxi Delays and Engine-Run Times

Guarantee Fairness

Enhance Predictability and Reduce Plan Violation

Maintain Controllers' Workload

Ensure Separation Standards

Airport System Objectives

Examples of Objective Functions

- Throughput: $q(\mathbf{t}) = \max(\mathbf{t})\, or \;\; q(\mathbf{t}) = \sum_{j=1}^{n} \tau_j^{\,p}, \quad p > 1$

where: $\tau_j = t_j - t_0$ is the relative time and t_0 is the present time.

- Taxi Times: $q(\mathbf{t}) = \sum_{i=1}^{n} c(\tau_j)\, \delta_{T,j}(t_j)^p$

where: $\delta_{T,j}(t_j) = t_j - t_{RTO,j}$ is the taxi delay for aircraft *j*.

- Fairness: $q(\mathbf{t}) = \sum_{i=1}^{n} \delta_{PB,j}(t_j)^p, \quad p > 0$

where: $\delta_{PB,j}(t_j) = \begin{cases} 0 & t_{PPB,j} < t_0 \\ t_{PPB,j}(t_j) - t_{RPB,j} & \text{otherwise} \end{cases}$ is the PB delay for departure *j*.

Types of Constraints

- Hard constraints
  - must be satisfied by all solutions
    - ✓ exclusive use of a runway
    - ✓ wake vortex separations
- Weak constraints
  - can be violated
  - the smaller the violation the better the solution
    - ✓ scheduled takeoff (due to latest flight plan)
- When possible, use a weak constraint

Categories of Constraints

- General Operational Constraints
  - Safety constraints
    - ✓ wake vortex separations
    - ✓ need translation in Departure-Departure-Separation (DDS), ADS, DAS
  - Miles (Min) In Trail due to SID structure
  - Runway usage rules
- Specific Operational Constraints
  - Controller inputs
  - New or changed constraints, e.g. EDCT or limiting swaps
- Physical Constraints
  - operational and /or traffic model
    - ✓ taxi out times
    - ✓ runway crossings

Careful “grouping” (sequencing) can minimize the potential service rate reduction (due to wake vortices)

Sequence is affected

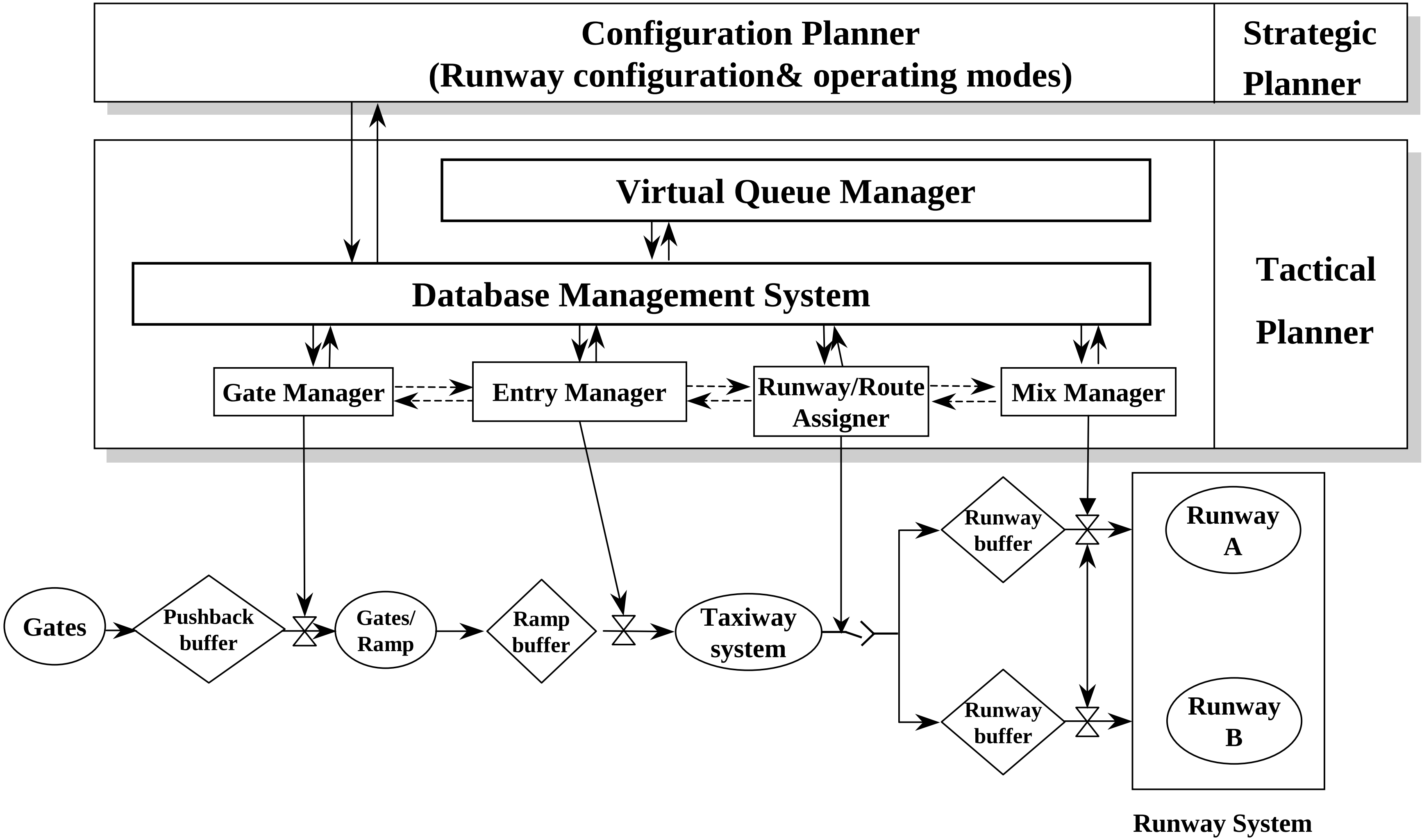
Possible propagation back to other control points

Examples of Constraints

- Wake Vortex
  - Adjacent Relative Position: $|x_i - x_j| < n$, $\forall i,j$ in the same aircraft class, n = number of aircraft of that class
- Distance or Time translated to relative position constraints
  - Miles In Trail
  - Minutes In Trail

  $\left.\begin{array}{l}\text{- Miles In Trail}\\ \text{- Minutes In Trail}\end{array}\right\} |x_i - x_j| \geq \Delta p$, p = TO sequence number
- Time translated to fixed position constraints
  - EDCT: $x_i = p_1$; $p_1 \leq x_i \leq p_2$
  - Lifeguard flights: $x_i = 1$; $1 \leq x_i \leq p_3$
  - Other priorities: $x_i < x_j$
- Deviation from First Come First Serve
  - Maximum Takeoff Position Shifting: $|x_{PB} - x_{TO}| \leq \Delta p_{limit}$

System Architecture



System Architecture: Research Topics

- Validate the proposed architecture at other airports with different characteristics, e.g. EWR with more hub operations than Logan
- Component role definition and interaction with other tools (EDP, FAST)
- Identification of component inputs
  - Define information exchange & constraint propagation between components
  - Link tactical subcomponents to the VQM and the Database Mgt System
- Queuing control
  - How does performance (e.g. throughput, delays, noise and emissions) change as the number of “look-ahead” queues increases (feedback)
  - How is performance if you have a “master” queue (system performance limits)
  - How is performance affected if downstream controllers (subcomponents) have knowledge of what is happening upstream (feed-forward)

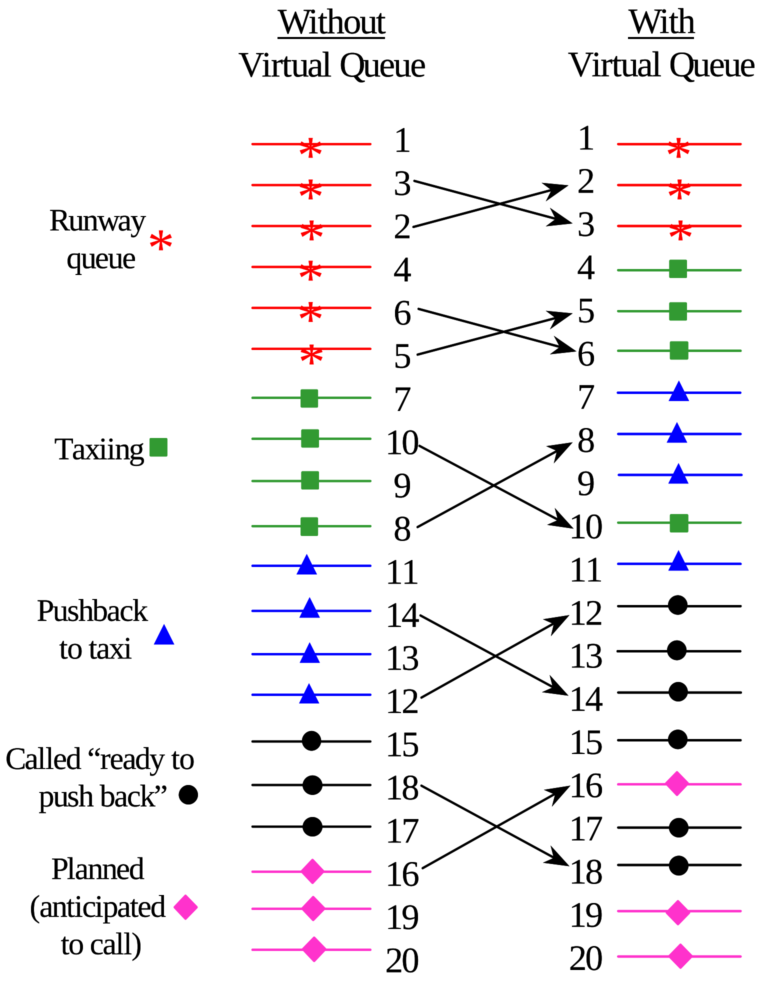
Virtual Queue Manager

- Central processing function that coordinates other tactical DP elements
- Possible modes of interaction between VQM and other tactical DP elements:
  - VQM performs global optimization and assigns regulatory tasks to other tactical DP elements
  - DP elements perform their own local optimization and VQM combines all the “local” solutions into a “global” solution
- Challenge is to design the Virtual Queue so that aircraft queues in the system (especially the runway takeoff queue) are neither “starved” nor “saturated”
- Possible virtual queue may have “physical” part that resides at the runway threshold and “virtual” part that includes all flights being considered in optimization
  - Flights in “physical” part are “frozen” a few (10 or 15) minutes before their assigned takeoff times
  - Flights in “virtual” part have scheduled departure times and sequence positions that are subject to change

Virtual Queue: Example

- Left side represents First Come First Serve transition from one state to the next where the queue buffer sizes are not controlled
  - results in unnecessarily overloaded takeoff queues and taxiway congestion

- Right side represents Virtual Queue implementation that controls number of aircraft in each state at each point in time and regulates timing of aircraft transitions from one state to the next



Virtual Queue: Possible Management Strategies

- Queue Size (N) Control
  - Number of aircraft in the system (after pushback) or in the takeoff queue
  - Easily monitored through flight strips
  - Gate vs. takeoff queue delays raises gate capacity issues
- Sequence - Based: control takeoff order while time is flexible
  - Key system constraints translated to sequence position constraints
  - Heuristic rules used to address certain types of inefficiencies
  - Only pushback order control
    - minimal disruption to controllers
    - suffers from taxi-out time uncertainty
  - Control via sequence adjustments at other control points
    - provides robustness against taxi time uncertainty
    - may increase controller workload
  - Control actions at each control point may depend on upstream / downstream control decisions
- Time - Based: control time slot allocation which determines the sequence
  - May be combined with sequence-based heuristic rules

Virtual Queue: Design and Implementation

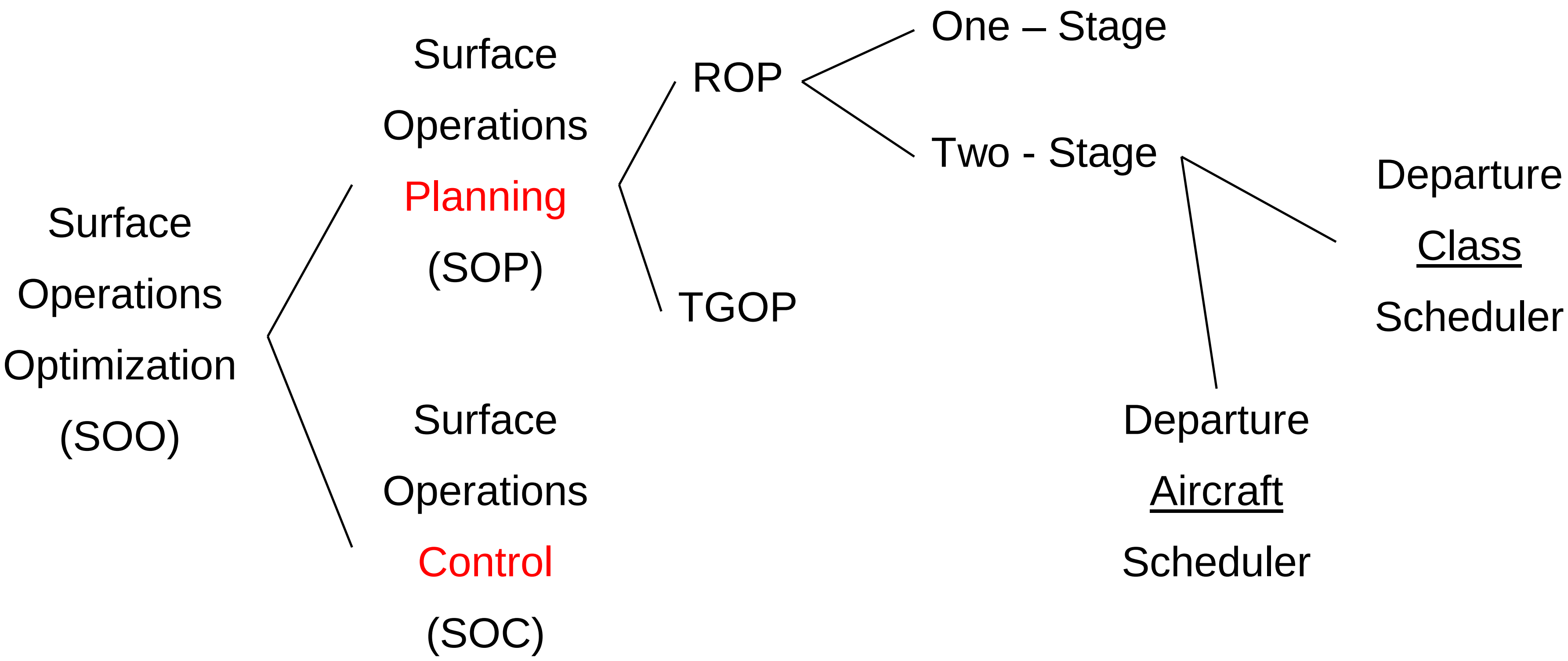
- Optimization triggering mechanisms
  - Event based
    - New “call ready for pushback”
    - Inability to takeoff (restricted runway access)
    - New arrival / crossing
  - Time based
    - Optimization time step (e.g. 2 minutes)

- Decision mechanism for committing aircraft to a specific plan (configuration dependent)
  - Sequence based: what part to be excluded (“frozen”) from the next optimization
  - Location based, e.g. “freeze” horizon beyond a certain control point
  - Time based: “freeze” horizon within a certain period from expected wheels off time

- Solution Quality
  - Stability vs. Optimality trade-offs
  - Possibly need a pseudo-physical way to talk about trades and communicate them to system operators (controllers)

- Human Factors Issues
  - Type and quantity of plan information presented to each controller position
  - Real time optimization functions vs. offline evaluation of generated plans

Surface Operations Optimization: Problem Structure


Surface Operations Optimization (SOO)
Surface Operations Planning (SOP)
Surface Operations Control (SOC)
ROP
TGOP
One – Stage
Two - Stage
Departure Class Scheduler
Departure Aircraft Scheduler

- SOP: Develop feasible and optimal departure plans that achieve desired objectives
- SOC: Execute plans

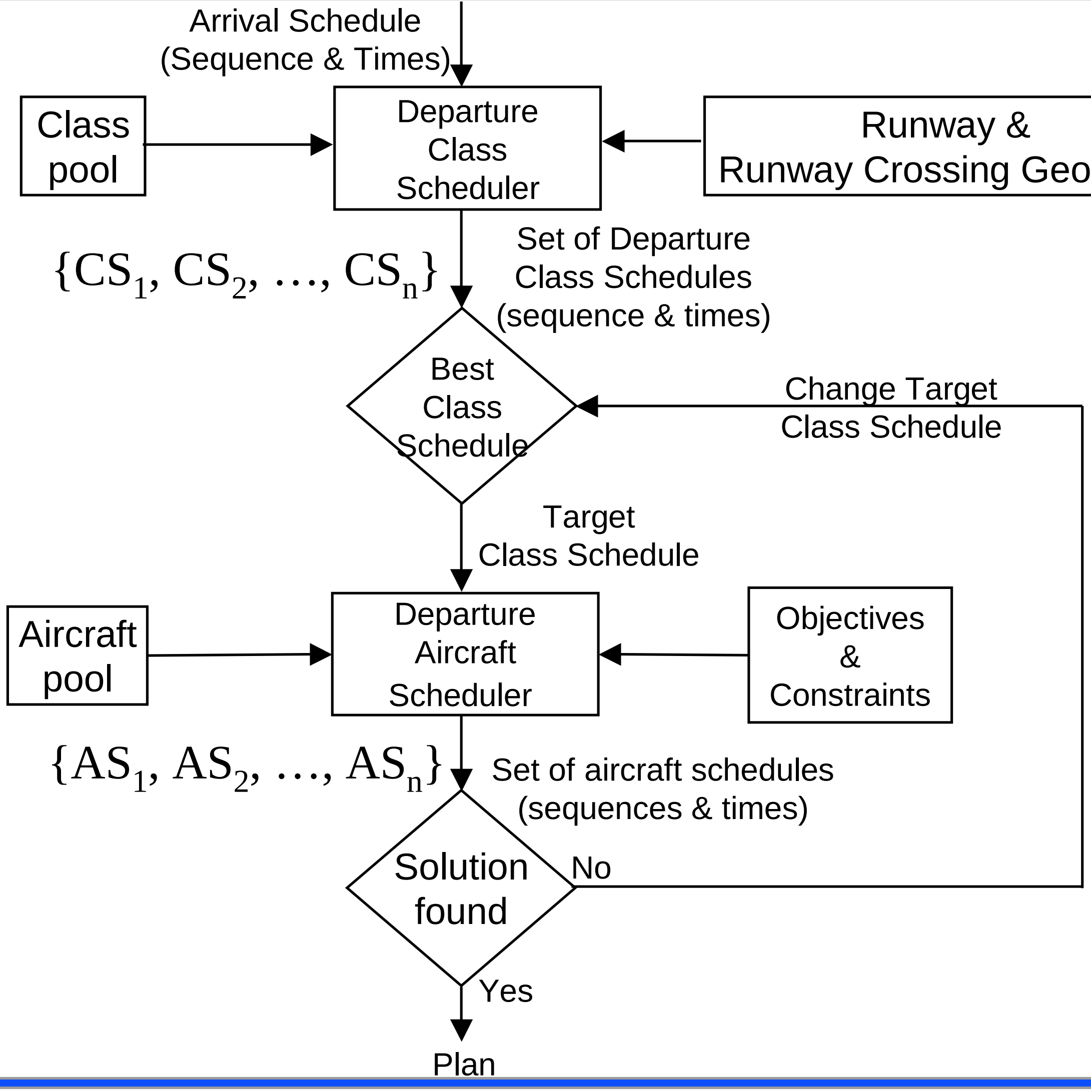
Surface Operations Planning

- Runway Operations Planning (ROP): Generate plan that achieves desired objectives
  - Single objective: Plans based on different objectives may be contradictory
  - Multi-objective: Planning specific takeoff times becomes too complex while planning only sequences is not enough
  - Time can relate different objectives in the same function and must be considered in the solution methodology
  - May be done in a single stage (AIAA paper) or in two stages (presented here)

- Taxi and Gate/Ramp Operations Planning (TGOP)
  - Important step, influences the feasibility of ROP
  - Back - propagate the runway plan to generate related plans for other airport locations (Gates, Ramps, Taxiways) and the associated DP components (GM, EM, RRA)
  - Possibly add buffers for uncertainty

MIT ICAT

Runway Operations Planning in Two Stages


Arrival Schedule
(Sequence & Times)
Class pool
Departure Class Scheduler
Runway & Runway Crossing Geometry
$\{CS_1, CS_2, \ldots, CS_n\}$
Set of Departure Class Schedules (sequence & times)
Best Class Schedule
Change Target Class Schedule
Target Class Schedule
Aircraft pool
Departure Aircraft Scheduler
Objectives & Constraints
$\{AS_1, AS_2, \ldots, AS_n\}$
Set of aircraft schedules (sequences & times)
Solution found
No
Yes
Plan

Taking advantage of the reduced uncertainty in the predicted demand for departure resources in terms of weight class relative to the predicted demand for departure resources in terms of specific aircraft

Runway Operations Planning in Two Stages (cont.)

- Taking advantage of the reduced uncertainty in the predicted demand for departure resources in terms of weight class relative to the predicted demand for departure resources in terms of specific aircraft

- Stage 1: Departure Class Scheduling
  - Assume fixed arrival schedule
  - Generate departure class schedules CS = {CS_1, … , CS_i, … , CS_m} based on a SINGLE objective (max. throughput) and the runway geometry

- Stage 2: Departure Aircraft Scheduling
  - For the Target Class Sequence optimize other objectives by assigning specific flights of the pre-assigned weight class to each departure class slot, a matrix AS of aircraft schedules is generated AS = {AS_1, … , AS_j, … , AS_n}

Simulation Approach: Simulink Model Design

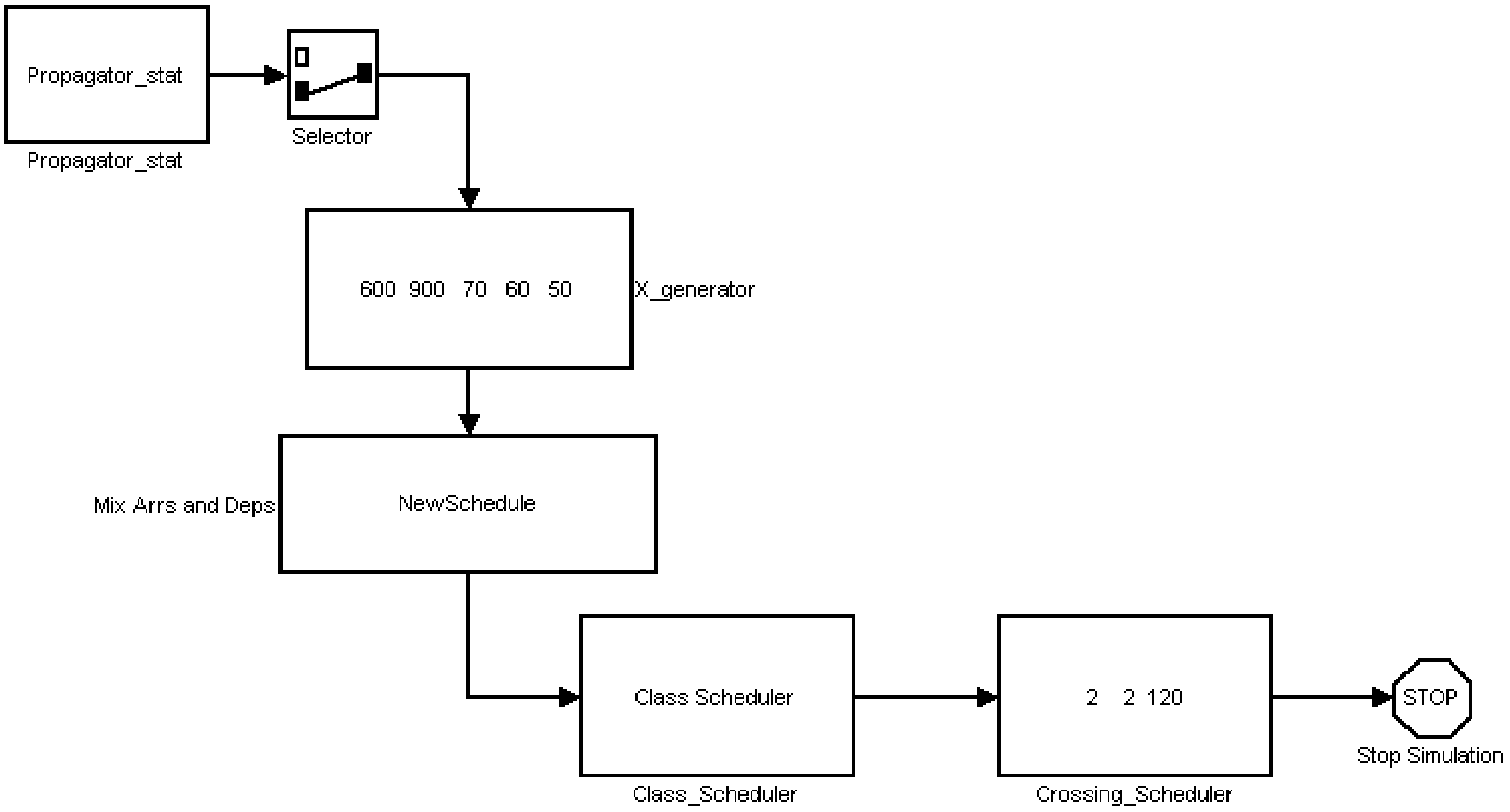
- Planning & control mechanisms to mitigate identified inefficiencies
  - Single resource
    - ✓ Clearance vs. Holding
    - ✓ Sequencing
    - ✓ Routing
  - Interaction between multiple resources
    - ✓ Blocking
- Model compatible with the proposed architecture
  - Airport components
  - Control points
  - Tactical planner

Simulation Approach: Simulink Model Design (cont.)

- To model the problem as a control system we need:
  - A plant (airport system)
  - A controller (departure planner decision aid tool): **under preparation**

- Model characteristics
  - Modularity
  - Internal vs. External control logic
  - Nested control loops: external vs. internal inputs

Full Model


Propagator_stat
Propagator_stat
Selector
600 900 70 60 50
X_generator
Mix Arrs and Deps
NewSchedule
Class Scheduler
Class_Scheduler
2 2 120
Crossing_Scheduler
STOP
Stop Simulation

Model Components

- Schedule files
- Preprocessor
  - Predetermined planning period
  - Develops all possible class schedules for the aircraft classes in hand (permutations)
  - Eliminates repetitions from the list of schedules
- Propagator
  - Propagates “dummy” aircraft through the airport model
  - Generates file with “propagation” values (minimum taxi times)
  - Modular, can model any airport layout quickly

Schedule Files

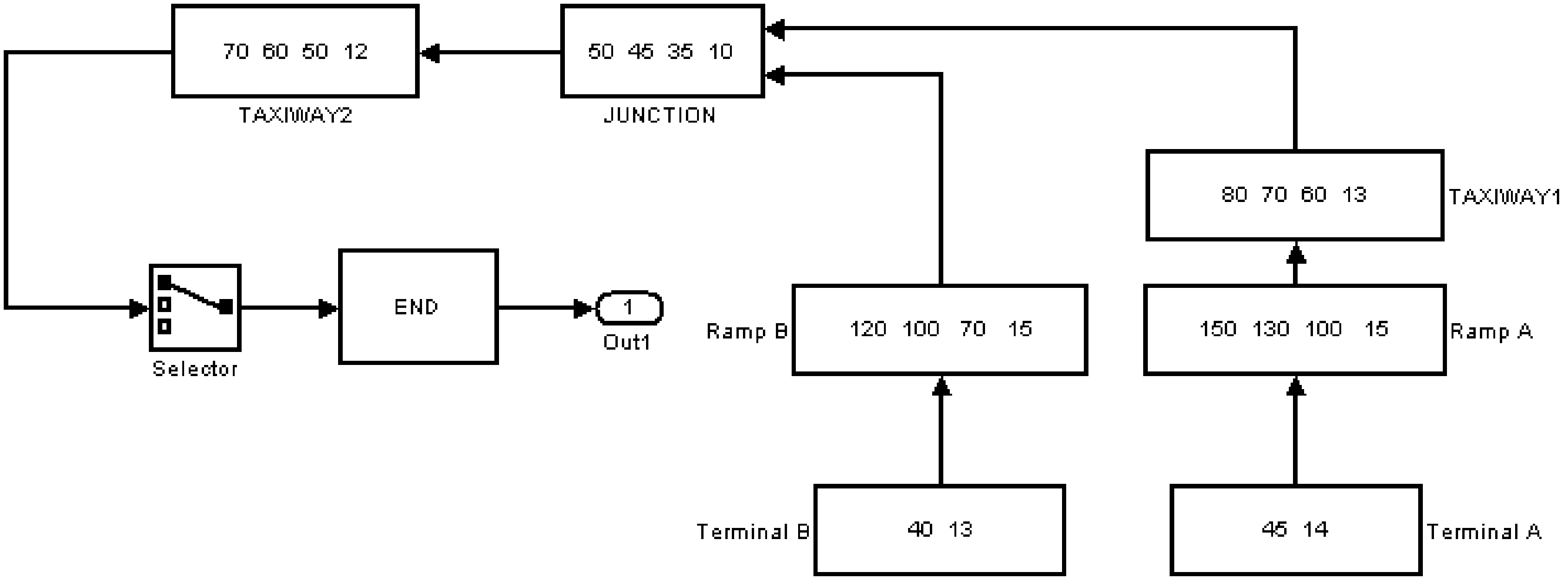
```
DEPAC(1).Simulation.Source='Terminal A';
DEPAC(1).Parameter.Type='J';
DEPAC(1).Parameter.Class='L';
DEPAC(1).Parameter.Model='B737';
DEPAC(1).FlightPlan.Destination='ATL';
DEPAC(1).Simulation.EntryTime=[1];
DEPAC(1).Identification.CallSign='DL182';
DEPAC(2).Simulation.Source='Terminal A';
DEPAC(2).Parameter.Type='J';
```

```
ARRAC(1).Simulation.Source='Fix A';
ARRAC(1).Parameter.Type='J';
ARRAC(1).Parameter.Class='H';
ARRAC(1).Simulation.LandingTime=[2];
ARRAC(2).Simulation.Source='Fix A';
ARRAC(2).Parameter.Type='P';
```

Runway Operations Planning Problem

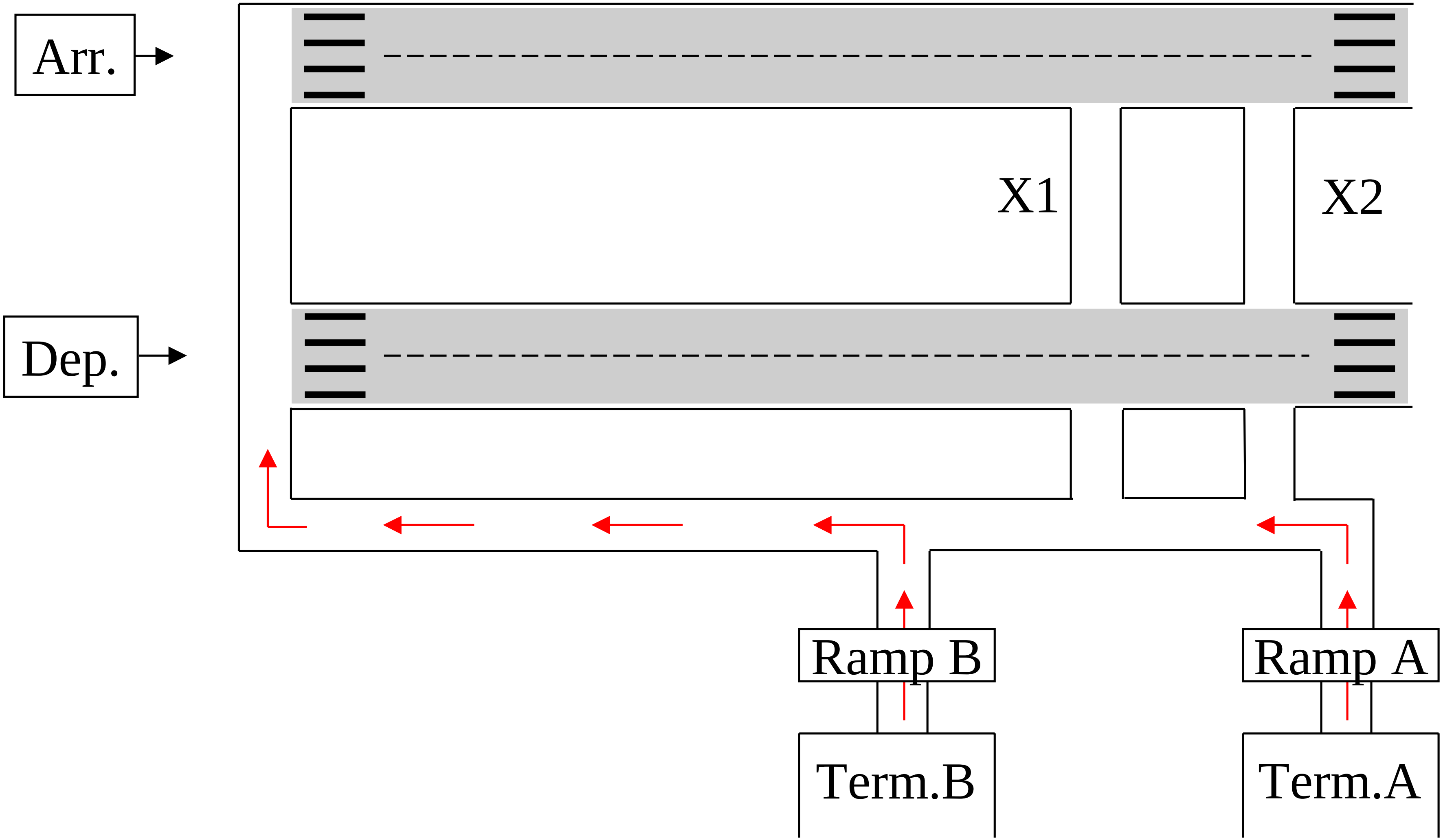
- Problem Description
  - Two parallel runways (one for arrivals and one for departures)
  - Taxiway space for holding aircraft between the two runways
- Assumptions
  - An arrival sequence is given and cannot be altered
  - Maximum number of aircraft allowed between runways is predetermined (test parameter)
  - Only three aircraft classes considered
  - Predetermined runway and taxiway occupancy times and separation criteria
- Objectives
  - Determine optimal schedule of operations between the two runways (arrivals, departures and crossings)
  - Determine optimal weight class schedules
  - Investigate specific weight class dependent patterns and possibly develop specific heuristic rules for different traffic scenarios

Propagator



- Numerical values are the average and standard deviation values for the time it takes a Heavy / Large / Small aircraft to traverse the specific airport component

Airport Model



Model Subsystems (cont.)

- X_generator
  - Looks into the dep. schedule and based on the propagation values, “propagates” departing aircraft to the runway threshold
  - Looks into the arr. schedule and based on expected touch-down times, calculates when each arrival is expected to become a crossing request
- NewSchedule
  - Mixes departure schedule and arrivals/crossing requests to a Mixed Schedule for the departure runway
- Class_Scheduler
  - Generates departure class schedules not yet including time for crossing requests
- Crossing_Scheduler
  - Prepares and introduces crossings in the schedule

Class_Scheduler & Crossing_Scheduler

- Starting point: the departure class schedules generated without crossings at preprocessing
- Crossing “gaps” introduced based on
  - Predetermined objectives (e.g. throughput related)
  - Heuristic rules
  - System constraints
- Throughput vs. Robustness trade-offs
- “Smart crossings”

Class_Scheduler & Crossing_Scheduler (cont.)

Schedule										Throughput
Index No										
30	L	L	H	L	L	L	S	S	H	540
45	L	L	H	S	S	L	L	L	H	540
63	L	L	L	H	S	S	L	L	H	540
90	L	L	L	H	L	L	S	S	H	540
105	L	L	L	S	S	H	L	L	H	540
3	L	L	H	L	S	S	L	L	H	540
22	L	L	H	L	L	S	S	L	H	540
41	L	L	H	S	L	L	L	S	H	540
71	L	L	L	H	S	L	L	S	H	540
82	L	L	L	H	L	S	S	L	H	540

Schedule															Throughput	
Index No															Without X	With X
30	L	L	s	s	h	H	s	l	L	L	L	S	S	H	540	590
45	L	L	s	s	h	H	s	l	S	S	L	L	L	H	540	590
63	L	L	s	s	h	L	s	l	H	S	S	L	L	H	540	630
90	L	L	s	s	h	L	s	l	H	L	L	S	S	H	540	630
105	L	L	s	s	h	L	s	l	S	S	H	L	L	H	540	630
3	L	L	s	s	h	H	s	l	L	S	S	L	L	H	540	590
22	L	L	s	s	h	H	s	l	L	L	S	S	L	H	540	590
41	L	L	s	s	h	H	s	l	S	L	L	L	S	H	540	590
71	L	L	s	s	h	L	s	l	H	S	L	L	S	H	540	630
82	L	L	s	s	h	L	s	l	H	L	S	S	L	H	540	630

708	S	H	L	L	S	L	L	L	H	540
716	S	H	L	S	L	L	L	L	H	540
687	S	L	S	L	H	L	L	L	H	540
756	S	S	L	L	L	L	L	H	H	510
46	L	L	H	S	S	H	L	L	L	600

708
716
687
756
46

Not practical

Runway Crossing Problem: Definition

- Geometry:
  - Two parallel runways, one for arrivals, one for departures
  - After exiting the arrival runway, aircraft have to cross the departure runway
- Objectives
  - “Group” runway crossing aircraft
  - Preferably grant crossing clearances behind a “heavy” departure (if there is one)
- Constraints
  - Maximum wait for any aircraft that requests crossing clearance does not exceed a pre-specified tolerable limit (e.g. 8 minutes)
  - Average wait for all aircraft cleared to cross at any given time is below a pre-specified limit (e.g. 10 minutes)

Departure & Crossing Process: Objectives, Rules, Constraints

- Maximize or “not decrease” throughput
  - Minimize # of weight class transitions
  - Small (S) before Heavies (H)
  - Large (L) before Heavies (H)
- Minimize departing aircraft “position shifting”
- If there is a Heavy (H), (if possible) place it before the point at which you wish to / must perform crossings

----

- Maximum individual delay
- Earliest time you can have an aircraft of a specific class at the runway end (physical constraint)

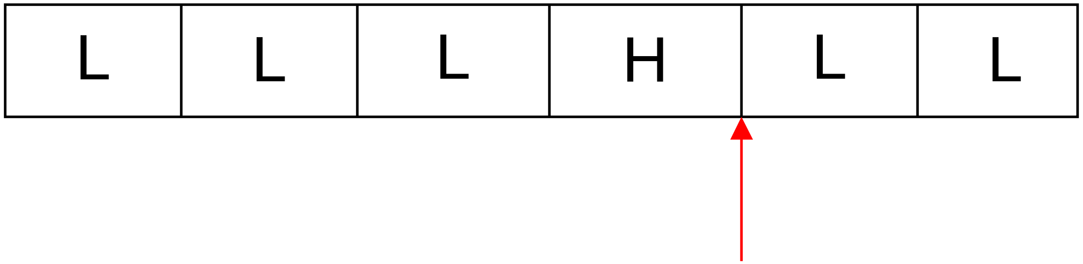
Trade-off:
Throughput vs. Robustness

- Maximize throughput : Heavy (H) at the end

L	L	L	L	L	H

6 a/c in 300 sec (5*60 sec)

- Most robust : Earlier availability of a crossing gap at the expense of throughput

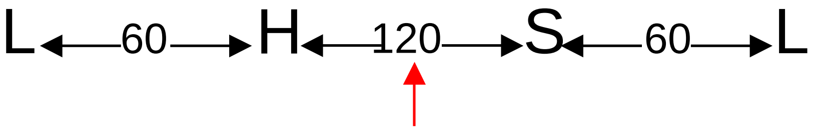


L	L	L	H	L	L

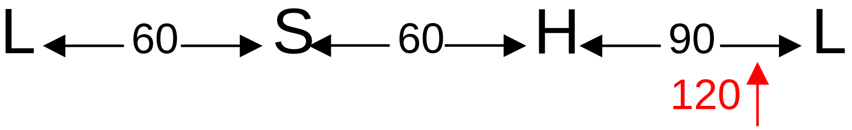
6 a/c in 330 sec (4*60 + 90 sec)

Smart Crossings

- Assumptions:
  - Three aircraft are waiting to cross
  - Crossing time needed = 40 + 10 + 10 = 60 sec
  - Roll time for departing aircraft before crossings = 60 sec
  - Total gap duration necessary = 60 + 60 = 120 sec
- Need to cross after a Heavy in the departure sequence
- Smart departure sequencing can maximize throughput
- Example: two Large (L), one Heavy (H) and one Small (S)



- Without crossings: 240 sec
- With crossings in an H-S gap, also 240 sec



- Without crossings: 210 sec
- With crossings in an H-L gap, 240 sec (need additional 30 sec)

Smart Crossings (cont.): Link arrival sequencing to DP

- Inputs
  - Arrival classes
  - Crossing point capacities (between the parallel runways)
  - Assume all Small go to one crossing point and all Large and Heavies to another (may be relaxed)
- Output
  - Smart departure AND arrival sequencing can maximize throughput
  - Find the best arrival sequence to bring arrivals to the crossing points so that no crossing point capacity is wasted due to saturation of another crossing point
- Example (both cross point capacities = 2)
  - Arr. sequence: S(210, 0.5) - S(240, 0.5) - L(270, 1) - S(290, 0.5) – H(340, 1.5)
  - Cross point 1 (all Small): S(210, 0.5) - S(240, 0.5) - S(290, 0.5) – TOTAL=1.5
  - Cross point 2 (Large and Heavies): L(270, 1) - H(340, 1.5) – TOTAL=1
  - Putting L ahead of H saturates point 2 earlier and therefore wastes capacity at point 1 (1.5 < 2) and point 2 (1 < 2)
  - Better arrival sequence: S(1) – S(2) – H(3) – S(4) – L(5)
    - ➢ Cross point 1: S(1) – S(2) – L(5) TOTAL = 2
    - ➢ Cross point 1: H(3) – S(4) TOTAL = 2

Crossing Problem: Issues

- Solution may be brittle
  - If we fix on a specific pattern to be used, there is no robustness to uncertainty
  - May need to plan alternative “crossing plans” when solving the ROP
- Fairness
  - How long is it fair for an aircraft to be held before cleared to cross ?
  - What if aircraft has NO gate ?

Surface Operations Control

- Fully De - Centralized Control:

  - Each component acts independently

  - May lead to conflicting control actions by components and thus instability

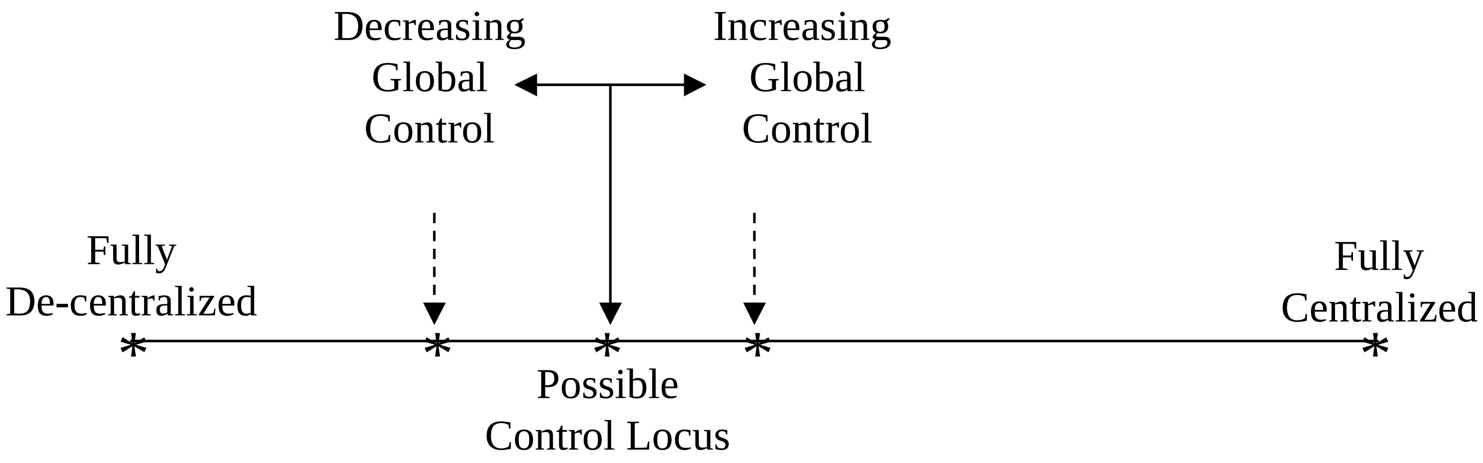
- Fully Centralized Control:

  - One central authority executes a plan for all aircraft and all positions

  - Requires full co-operation and information exchange among all system components

  - May become too complex for practical implementation

Surface Operations Control: Current Status



- In current operations
  - Controllers maintain a certain level of feedback and feed-forward communication in order to:
    - ➢ Gain an understanding of the situation upstream and downstream
    - ➢ Use it to determine the control actions they are responsible for
- Data analysis is necessary to understand how controllers operate and communicate with each other for feedback and feed-forward

Summary

- Departure Planner is intended to assist short-term planning operations at major commercial airports
    - Emphasis on supporting Air Traffic Management in the next 30 to 45 min from current time but also has strategic component that plans with a time-horizon of a few hours
- Consists of a set of functional components
    - Strategic configuration planning
    - Tactical departure planning
- Not necessarily fully automated system
    - Components could potentially become automation tools used by the controllers to manage the various physical queues existing in the flow of departing aircraft, without increasing workload levels.
- Design, implementation and potential benefit margins are very dependent on the specific airport structure and operational procedures.
- Airlines should participate in the “planning / implementation” loop: Need to outline a feasible approach path to get airlines interested and involved

Future Steps

- 1st Stage
  - Stochastic throughput calculations
  - “Smart crossings”
- 2nd Stage
  - Specific aircraft positioning based on other constraints
  - “Look-ahead” beyond the current aircraft pool
  - Throughput vs. Robustness trade-offs